TITLE
SYSTEM AND METHOD FOR HUFFMAN SHAPING IN A DATA
COMMUNICATION SYSTEM

CROSS-REFERENCE TO RELATED APPLICATIONS

This application makes reference to, and claims priority
to and the benefit of, United States provisional application Serial
No. 60/224,733 filed August 11, 2000.

INCORPORATION BY REFERENCE
The above-referenced United States provisional
application Serial No. 60/224,733 is hereby incorporated herein
by reference in its entirety.

STATEMENT REGARDING FEDERALLY SPONSORED
RESEARCH OR DEVELOPMENT

N/A

BACKGROUND OF THE INVENTION
Current data communication systems rarely approach
highest possible rate, i.e., the rate corresponding to Shannon
channel capacity. For example, voiceband modems complying
with ITU –T recommendation V.90 employ uncoded modulation
for downstream transmission. The nominal downstream rate of
56 kbit/s is thereby almost never achieved, although under
practical channel conditions the capacity rate can exceed 56
kbit/s.

The difference between the signal-to-noise ratio (SNR)
required to accomplish a given rate with a given practical coding
and modulation scheme and the SNR at which an ideal capacity-

achieving scheme could operate at the same rate is known as "SNR gap to capacity". At spectral efficiencies of 3 bit per signal dimension or higher, uncoded modulation with equiprobable PAM (pulse amplitude modulation) and QAM (quadrature amplitude modulation) symbols exhibit an SNR gap of 9 dB at a symbol error probability of 10^{-6}. In the case of V.90 downstream transmission, the SNR gap can correspond to a rate loss of up to 12 kbit/s.

This overall 9 dB gap is generally comprised of a "shaping gap" portion and a "coding gap" portion. The "shaping gap" portion (approximately 1.5 dB) is caused by the absence of constellation shaping (towards a Gaussian distribution). The remaining "coding gap" portion (approximately 7.5 dB) stems from the lack of sequence coding to increase signal distances between permitted symbol sequences.

Two different techniques are used, generally in combination, to reduce the overall 9 dB gap. The first technique addresses the "coding gap" portion, and uses one of several coding techniques to achieve coding gains. One of these techniques is trellis-coded modulation. More recent techniques employ serial- or parallel-concatenated codes and iterative decoding (*Turbo coding*). These latter techniques can reduce the coding gap by about 6.5 dB, from 7.5 dB to about 1 dB.

Once a coding gain is achieved, the second technique, referred to as shaping, can be used to achieve an even further gain. This type of gain is generally referred to as a shaping gain. Theoretically, shaping is capable of providing an improvement (i.e., shaping gain) of up to 1.53 dB.

Two practical shaping techniques have been employed in the prior art to achieve shaping gains, namely, trellis shaping and shell mapping. With 16-dimensional shell mapping, such as

employed in V.34 modems, for example, a shaping gain of about 0.8 dB can be attained. Trellis shaping can provide a shaping gain of about 1 dB at affordable complexity. Accordingly, between 0.5 and 0.7 dB of possible shaping gain remains untapped by these prior art shaping methods.

Further limitations and disadvantages of conventional and traditional approaches will become apparent to one of skill in the art, through comparison of such systems with the present invention as set forth in the remainder of the present application with reference to the drawings.

BRIEF SUMMARY OF THE INVENTION

Aspects of the present invention may be found in a method of communicating data in a communication system. The method generally comprises accepting and randomizing (scrambling) data from a source of user data, such as a computer, for example. The randomized data are accumulated until a Huffman codeword is recognized, at which time the Huffman codeword is mapped into a channel symbol. Then the channel symbol is applied to an input of a communication channel. In the field of source coding, the above operation is known as Huffman decoding.

The encoding operation described above may be combined with further channel encoding operations such as, for example, trellis coded modulation or some form of serial- or parallel-concatenated coding to achieve coding gain in addition to shaping gain. In addition, channel symbols can be modulated in various ways before they are applied to the input of the communication channel.

In one embodiment of the invention, the channel encoding operation described above is performed in combination with a framing operation to achieve transmission of data at a constant rate.

Next, on the receiver side of the communication channel, a channel symbol is received from an output of the communication channel after suitable demodulation and channel decoding. Once obtained, the channel symbol is converted into the corresponding Huffman codeword. The data sequence represented by concatenated Huffman codewords is de-randomized (descrambled) and delivered to a sink of user data.

In one embodiment of the invention, a deframing operation is performed, which provides for data delivery to the data sink at constant rate.

The method of the present invention results in a symbol constellation and a probability distribution of symbols in this constellation that exhibits a shaping gain of greater than 1 dB. The shaping gain may be, for example, 1.35 dB or 1.5 dB, depending on the specific design

In general, a communication system according to the present invention comprises a communication node that performs a "Huffman decoding" operation to generate channel symbols with a desired probability distribution.

These and other advantages and novel features of the present invention, as well as details of an illustrated embodiment thereof, will be more fully understood from the following description and drawings.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS
OF THE DRAWINGS

Fig.1 is a block diagram of a generic communication system that may be employed in connection with the present invention.

Fig. 2 illustrates additional detail regarding the transmitters of Fig. 1 according to the present invention.

Fig. 3 shows shaping gain versus rate for PAM and QAM_{sq} constellations of different sizes, in accordance with the present invention.

Fig. 4 plots shaping gains versus rate for square and lowest-energy 1024-QAM constellations, in accordance with the present invention.

Fig. 5 depicts the mean and standard deviation of the rate in bit/dimension and the shaping gain accomplished for a nominal rate of R = 4 bit/dimension with QAM_{le} constellations of different sizes, in accordance with the present invention.

Fig. 6 illustrates a 128-QAM_{le} constellation with Huffman shaping for a nominal rate of 3 bit/dimension, in accordance with the present invention.

Fig. 7 illustrates one embodiment of a generic method for achieving constant rate and recovering from bit insertions and deletions.

Fig. 8 illustrates the probability of pointer overflow as a function of framing buffer size in accordance with the present invention.

Fig. 9 illustrates one embodiment of the design of a Huffman code in accordance with the present invention.

Fig. 10 is a block diagram of one embodiment of a communication system that operates in accordance with the method of present invention.

Fig. 11 is another embodiment of the design of a Huffman code in accordance with the present invention, when a framer/deframer is utilized.

Fig. 12 is a block diagram of another embodiment of a communication system that operates in accordance with the method of present invention, utilizing a framer/deframer.

Fig. 13 illustrates one operation of a system that employs Huffman shaping in accordance with the present invention.

DETAILED DESCRIPTION OF THE INVENTION

Fig.1 is a block diagram of a generic communication system that may be employed in connection with the present invention. The system comprises a first communication node 101, a second communication node 111, and a channel 109 that communicatively couples the nodes 101 and 111. The communication nodes may be, for example, modems or any other type of transceiver device that transmits or receives data over a channel. The first communication node 101 comprises a transmitter 105, a receiver 103 and a processor 106. The processor 106 may comprise, for example, a microprocessor. The first communication node 101 is communicatively coupled to a user 100 (e.g., a computer) via communication link 110, and to the channel 109 via communication links 107 and 108.

Similarly, the second communication node 111 comprises a transmitter 115, a receiver 114 and a processor 118. The processor 118, like processor 106, may comprise, for example, a microprocessor. The second communication node 111 is likewise communicatively coupled to a user 120 (again a computer, for example) via communication link 121, and to the channel 109 via communication links 112 and 113.

During operation, the user 100 can communicate information to the user 120 using the first communication node 101, the channel 109 and the second communication node 111. Specifically, the user 100 communicates the information to the first communication node 101 via communication link 110. The information is transformed in the transmitter 105 to match the restrictions imposed by the channel 109. The transmitter 105 then communicates the information to the channel 109 via communication link 107. The receiver 114 of the second communication node 111 next receives, via communication link 113, the information from the channel 109, and transforms it into a form usable by the user 120. Finally, the information is communicated from the

second communication node 111 to the user 120 via the communication link 121.

Communication of information from the user 120 to the user 100 may also be achieved in a similar manner. In either case, the information transmitted/received may also be processed using the processors 106/118.

Fig. 2 illustrates additional detail regarding the transmitters of Fig. 1 according to the present invention. The functions of transmitter 201 may be decomposed into those of a source encoder 203 and a channel encoder 205. Generally, the source encoder 203 is a device that transforms the data produced by a source (such as the user 100 or user 120 of Fig. 1) into a form convenient for use by the channel encoder 205. For example, the source may produce analog samples at a certain rate, such as, for example, 8000/s, as in a telephone application. The source encoder 203 then may perform the function of analog-to-digital conversion, converting each analog sample into an 8-bit binary code. The output of the source encoder 203 then would be a binary sequence of digits presented to the input of the channel encoder 205 at a rate of 8 x 8000 = 64,000 bit/s. The output of the source encoder 203 is passed to the channel encoder 205, where the data are transformed into symbols that can be transmitted on the channel. For example, the data may be transformed using pulse-amplitude modulation (PAM), whereby successive short blocks of data bits of length N are encoded as analog pulses having one of 2^N allowable amplitudes.

In most communication systems, the data presented to the channel encoder are assumed to be completely random. This randomness is normally assured by the inclusion of a scrambler designed into the system. In the previous example of PAM, random data would lead to each 2^N of the allowable amplitudes being equally likely. That is, each of them occurs with probability 2^{-N}. It turns out that employing equally likely pulse amplitudes leads to a small inefficiency in the use of

the power in the signal that is transmitted into the channel. In fact, as mentioned above, if the amplitude distribution can be made more nearly Gaussian, then up to 1.53 dB of transmitted power can be saved for the same level of error performance at the receiver.

Accordingly, a shaping function is provided in Fig. 2 by a shaper 207, which alters the statistical distribution of the values presented to modulator 209. Shaping the transmitted signal generally means controlling the distribution of transmitted signal values to make the signal appear more Gaussian in character. The shaper 207 comprises a Huffman decoder 211 and a mapper 213. The design of the Huffman decoder 211 depends upon the characteristics of the channel.

In the Huffman decoder 211, the sequence of scrambled binary data bits is parsed into Huffman codewords. The codewords are then mapped into modulation symbols. The Huffman code is designed to let the modulation symbols assume approximately a sampled Gaussian distribution.

Unlike trellis shaping or shell mapping, Huffman shaping is not a constant-rate-encoding scheme. Moreover, decoding errors can lead to bit insertion or deletion in the decoded binary data sequence. This may be acceptable for many systems, such as, for example, those in which variable-length packets are transmitted in burst mode with an Ethernet-like medium access protocol. In some cases, continuous transmission at constant rate is desirable, such as, for example, those involving variable-rate encoded voice and video streams over constant rate channels. A constant rate and recovery from bit insertions and deletions may be achieved, and the framing overhead may be kept to a value equivalent to a SNR penalty of ≈0.1 dB, for example, utilizing the method of the present invention.

The following mathematical foundation of Huffman shaping is based upon M-ary PAM data transmission, but the

concept clearly applies to two- and higher-dimensional modulation as well.

Let A_M be a symmetric M-ary PAM constellation of equally spaced symbols. Adjacent symbols are spaced by 2, and M may be even or odd (usually M will be even):

(1)
$$A_M = \{a_i = -(M-1) + 2i, \ 0 \le i \le M-1\}$$
$$\text{e.g.:} \quad A_8 = \{-7,-5,-3,-1,+1,+3,+5,+7\}, \quad A_5 = \{-4,-2,0,+2,+4\}$$

If symbols are selected independently with probabilities $\mathbf{p} = \{p_i, \ 0 \le i \le M-1\}$, the *symbol entropy* $H(\mathbf{p})$ (= rate) and the *average symbol energy* $E(\mathbf{p})$ become:

(2)
$$H(\mathbf{p}) = -\sum_{i=0}^{M-1} p_i \log_2 p_i \quad \text{bit/symbol}$$
$$\left(p_i = \frac{1}{M} \forall i : H(\mathbf{p}) = H_M = \log_2 M \right),$$

(3)
$$E(\mathbf{p}) = \sum_{i=0}^{M-1} p_i \, |a_i|^2$$
$$\left(M-PAM, \ p_i = \frac{1}{M} \forall i : E(\mathbf{p}) = E_M = \frac{M^2 - 1}{3} \right).$$

Shaping gain $G_s(\mathbf{p})$ expresses a saving in average symbol energy achieved by choosing symbols from A_M with probabilities $\mathbf{p}$ rather than selecting equiprobable symbols from a smaller constellation $A_{M'}$, where $M' = 2^{H(\mathbf{p})}$ ($M' < M$, ignoring that M' may not be an integer):

(4)
$$G_s(\mathbf{p}) = \frac{E_{M'}}{E(\mathbf{p})} = \frac{2^{2H(\mathbf{p})} - 1}{3 \times E(\mathbf{p})}.$$

The maximum shaping gain is obtained by the probability distribution $\mathbf{p} = \widetilde{\mathbf{p}}$, which minimizes $E(\mathbf{p})$ subject to the constraints $R = H(\mathbf{p})$ and $\sum_{i=0}^{M-1} p_i = 1$. Differentiation of

$$(5) \qquad J(\mathbf{p}) = \sum_{i=0}^{M-1} p_i |a_i|^2 + \lambda_1 \left(-\sum_{i=0}^{M-1} p_i \log_2 p_i - R \right) + \lambda_2 \left(\sum_{i=0}^{M-1} p_i - 1 \right)$$

with respect to the probabilities p_i yields the conditions

$$(6) \qquad \left. \frac{\partial J(\mathbf{p})}{\partial p_i} \right|_{\mathbf{p}=\widetilde{\mathbf{p}}} = |a_i|^2 - \frac{\lambda_1}{\ln 2}\left(\ln \widetilde{p}_i + 1 \right) + \lambda_2 = 0, \quad \text{for } 0 \le i \le M-1.$$

The parametric solution of (6), with the Lagrange multipliers λ_1, λ_2 transformed into the new variables α, s, becomes

$$(7) \qquad \widetilde{p}_i = \exp\left(-1 + \frac{\ln 2}{\lambda_1}(|a_i|^2 + \lambda_2) \right) = \alpha \, \exp\left(-s \, |a_i|^2 \right) \ , \ 0 \le i \le M-1.$$

The optimum distribution $\widetilde{\mathbf{p}}$ is thus found to be a Gaussian distribution sampled at the symbol values of A_M . This solution can also be obtained by maximizing the rate $R = H(\mathbf{p})$ subject to the constraints $E(\mathbf{p}) = S$ and $\sum_{i=0}^{M-1} p_i = 1$. The value of α follows from $\sum_{i=0}^{M-1} p_i = 1$. The value of s may be chosen to achieve a given rate $R \le \log_2(M)$ or a given average symbol energy $S \le E_M$.

If M and R are increased, the optimum shaping gain tends towards the *ultimate shaping gain* $G_s^\infty = \pi e / 6 = 1.423$ (1.53 dB). This gain can be derived as the ratio of the variance of a uniform density over a finite interval and the variance of a Gaussian density, both with the same differential entropy.

One can see that (7) does not only hold for regular symmetric PAM constellations, but gives the optimum shaping

probabilities for arbitrary one- and higher-dimensional symbol constellations as well.

In general, given a sequence of M-ary source symbols which occur independently with probability distribution **p**, a traditional Huffman coding approach encodes the source symbols into binary codewords of variable lengths such that (a) no codeword is a prefix of any other codeword (*prefix condition*), and (b) the expected length of the codewords is minimized.

An optimum set of codewords is obtained by Huffman's algorithm. More particularly, let a_i be a source symbol that occurs with probability p_i. The algorithm associates a_i with a binary codeword c_i of length ℓ_i such that $2^{-\ell_i} \approx p_i$. The algorithm guarantees that $\sum_{i=0}^{M-1} 2^{-\ell_i} = 1$ (Kraft's inequality is satisfied with equality), and that the expected value of the codeword length, $L = \sum_{i=0}^{M-1} p_i \ell_i$, approaches the entropy of the source symbols within one bit [10]:

(8) $\qquad H(\mathbf{p}) \le L < H(\mathbf{p}) + 1$.

In the limit for large $H(\mathbf{p})$, the concatenated Huffman codewords yield a binary sequence of independent and equiprobable zeroes and ones with rate $R = L \cong H(\mathbf{p})$ bit per source symbol. However, for certain probability distributions L may be closer to $H(\mathbf{p}) + 1$ than $H(\mathbf{p})$ because of quantization effects inherent in the code construction. If $H(\mathbf{p})$ is small, the difference between L and $H(\mathbf{p})$ can be significant. The rate efficiency may be improved by constructing a Huffman code for blocks of $K > 1$ source symbols. Then, (8) takes the form $H(\mathbf{p}) \le L(K)/K = L \le H(\mathbf{p}) + 1/K$, where $L(K)$ is the expected

length of the Huffman codewords associated with K-symbol blocks. The code comprises M^K codewords and the rate expressed in bit per source symbol will generally be within 1/K bit from $H(\mathbf{p})$.

With the Huffman shaping method of the present invention, the traditional encoding approach is reversed. A Huffman code is generated for the optimum probability distribution $\tilde{\mathbf{p}}$ of the modulation symbols in a given M-ary constellation. In the transmitter, the sequence of data bits is suitably scrambled so that perfect randomness can be assumed. The scrambled sequence is buffered and segmented into Huffman codewords, as in traditional Huffman decoding. A codeword c_i is encountered with probability $2^{-\ell_i} \approx \tilde{p}_i$ and mapped into modulation symbol a_i. In the receiver, when a symbol a_i is detected codeword c_i is inserted into the binary output stream.

For the general case of K-dimensional modulation (K = 1: PAM, K=2: QAM), it is appropriate to express rates and symbol energies per dimension, while $a_i, \tilde{p}_i,$ and ℓ_i relate to K-dimensional symbols.

The mean value $\overline{R}^h$ and the standard deviation σ_R^h of the number of bits encoded per symbol dimension become

$$(9) \qquad R^h = \frac{1}{K} \sum_{i=0}^{M-1} 2^{-\ell_i} \ell_i \qquad \text{bit/dimension} \left(\approx \frac{1}{K} H(\tilde{p}) \right),$$

$$(10) \qquad \sigma_R^h = \sqrt{\frac{1}{K} \sum_{i=0}^{M-1} 2^{-\ell_i} (\ell_i - KR^h)^2} \ .$$

The average symbol energy per dimension S^h and the shaping gain G_s^h of the Huffman-shaped symbol sequence are given by

$$(11) \qquad S^h = \frac{1}{K} \sum_{i=0}^{M-1} 2^{-\ell_i} \mid a_i \mid^2 \quad \text{energy per dimension} \quad \left(\approx \frac{1}{K} E(\widetilde{p}) \right),$$

$$(12) \qquad G_s^h = \frac{2^{2\overline{R}^h} - 1}{3 \times E^h} \, .$$

The corresponding quantities obtained with optimum shaping probabilities $\widetilde{p}$ will be denoted, respectively, by $\widetilde{R}$ and $\sigma_{\widetilde{R}}$ (bit/dimension), $\widetilde{S}$ (energy per dimension), and $\widetilde{G}_s$ (optimum shaping gain).

For numerical evaluations, uncoded modulation with M-PAM (M=2m) and M-QAM (M=4m) constellations have been considered. The M-QAM constellations are either *square* constellations M-QAM$_{sq}$ = $\sqrt{M}$ -PAM $\times \sqrt{M}$ -PAM, or *lowest-energy* constellations M-QAM$_{le}$ comprising the M points in the set $\{(1+2i, 1+2k), \ i, k \in \mathbf{Z}\}$ nearest to the origin. The symmetries of the symbol constellations are enforced on the Huffman codes. In the PAM case, m codewords are constructed for positive symbols and then extended by a sign bit. Similarly, in the QAM case m codewords are constructed for symbols in the first quadrant and extended by two quadrant bits. The results of different numerical evaluations are depicted in Figs 3, 4, and 5.

Fig. 3 shows shaping gain versus rate for PAM and QAM$_{sq}$ constellations of different sizes, in accordance with the present invention. The solid curves indicate the shaping gains obtained

with the optimum shaping probabilities $\tilde{\mathbf{p}}$. Every rate in the interval $1 \leq R \leq \log_2(M)/K$ can be accomplished (bit per dimension). The shaping gains vanish at $R = 1$ (constellations reduced to BPSK or QPSK) and $R = \log_2(M)/K$ (equiprobable M-QAM). The optimum shaping gains practically reach the ultimate shaping gain of 1.53 dB at $R = 4$ bit per dimension for $\geq$32-PAM and $\geq$1024-QAM$_{sq}$ constellations. With the Huffman shaping method of the present invention, not every rate can be realized because of quantization effects in the construction of Huffman codes. For PAM, shaping gains of up to $\approx$1.35 dB are achieved at some rates above 3 bit per dimension. The effects of quantization are significantly reduced in the QAM cases. With $\geq$256-QAM$_{sq}$ constellations shaping gains within 0.1 dB from the ultimate shaping gain of 1.53 dB are consistently obtained at rates above 3 bit per dimension.

Fig. 4 plots shaping gains versus rate for square and lowest-energy 1024-QAM constellations, in accordance with the present invention. Minor differences occur in the region of diminishing shaping gains, at rates above 4.5 bit/dimension. The shaping gain of equiprobable 1024-QAM$_{le}$ ($R = 5$ bit/dimension) is 0.2 dB.

Fig. 5 depicts the mean and standard deviation of the rate in bit/dimension and the shaping gain accomplished for a nominal rate of $R = 4$ bit/dimension with QAM$_{le}$ constellations of different sizes, in accordance with the present invention. The nominal rate is at least closely achieved with Huffman shaping (with optimum shaping it is exactly achieved). The standard deviation increases with increasing constellation size to a final value of $\approx$1 bit/dimension. The optimum shaping gain and the

Huffman shaping gain increase rapidly when the initial 256-QAM constellation is enlarged. The respective final shaping gains of ≈ 1.5 dB and ≈ 1.4 dB are practically achieved with M = 512 (512-QAM$_{le}$: 1.495 dB and 1.412 dB, 1024-QAM$_{le}$: 1.516 dB and 1.432 dB).

Fig. 6 illustrates a 128-QAM$_{le}$ constellation with Huffman shaping for a nominal rate of 3 bit/dimension, in accordance with the present invention. The codeword lengths ranging from 5 to 12 bits are indicated for the first-quadrant symbols.

$\bar{R}^h = 2.975$ ($\sigma_R^h = 0.919$) bit/dimension and

$G_s^h = 1.378 \, \text{dB}$ ($\tilde{G}_s = 1.443 \, \text{dB}$) are achieved. The symbol energies, optimum shaping probabilities, codeword probabilities and lengths, and the codewords of the first quadrant symbols are listed below. The codewords for the first-quadrant symbols end with 00.

i	$|a_i|^2$	$\tilde{p}_i$	$p_i^h = 2^{-\ell_i}$	ℓ_i	c_i
0	2	0.03872	0.03125	5	00000
1	10	0.02991	0.03125	5	10000
2	10	0.02991	0.03125	5	01100
3	18	0.02311	0.03125	5	11100
4	26	0.01785	0.01563	6	010000
5	26	0.01785	0.01563	6	001100
6	34	0.01379	0.01563	6	110000
7	34	0.01379	0.01563	6	101100
8	50	0.00823	0.00781	7	1010000
9	50	0.00823	0.00781	7	0101100
10	50	0.00823	0.00781	7	0101000
11	58	0.00636	0.00781	7	1101100
12	58	0.00636	0.00781	7	1101000
13	74	0.00379	0.00391	8	10101100
14	74	0.00379	0.00391	8	10101000
15	82	0.00293	0.00195	9	001001000
16	82	0.00293	0.00195	9	001000100
17	90	0.00226	0.00195	9	001010100
18	90	0.00226	0.00195	9	001010000
19	98	0.00175	0.00195	9	001011100
20	106	0.00135	0.00098	10	0010011100
21	106	0.00135	0.00098	10	0010011000
22	122	0.00081	0.00049	11	00100000100

23	122	0.00081	0.00049	11	00100000000
24	130	0.00062	0.00049	11	00101101000
25	130	0.00062	0.00049	11	00101100100
26	130	0.00062	0.00049	11	00101100000
27	130	0.00062	0.00049	11	00100001100
28	146	0.00037	0.00024	12	001000010100
29	146	0.00037	0.00024	12	001000010000
30	162	0.00022	0.00024	12	001011011000
31	170	0.00017	0.00024	12	001011011100

Table 1 Huffman code words tabulated against their index

Fig. 7 illustrates one embodiment of a generic method for achieving constant rate and recovering from bit insertions and deletions. Data frames of N_b bits are embedded into symbol frames of N_s modulation symbols. Every sequence of bits transmitted within a symbol frame begins with a S&P (synch & pointer) field of $n_{sp} = n_s + n_p$ bits, where n_s is the width of a synch subfield and n_p is the width of a pointer subfield. The synch subfield enables the receiver to acquire symbol-frame synchronization. In principle, sending a known pseudo-random binary sequence with one bit ($n_s = 1$) in every S&P field is sufficient (as in T1 systems). The pointer subfield of the nth symbol frame expresses the offset in bits of the nth data frame from the S&P field.

With reference to Fig. 7, in the 1st symbol frame, the 1st data frame follows the S&P field with zero offset. The S&P field and 1st data frame are parsed into Huffman codewords, which are then mapped into modulation symbols indexed by 1,2,3, ... N_s. The end of the 1st data frame is reached before the N_s^{th} modulation symbol has been determined. The data frame is padded with fill bits until the N_s^{th} modulation symbol is obtained. The 2nd data frame follows the S&P field of the 2nd symbol frame again with zero offset. Now the last symbol of the 2nd symbol

frame is found before the 2nd data frame is completely encoded. The S&P field of the 3rd symbol frame is inserted and encoding of the remaining part of the 2nd data frame is then continued, followed by encoding the 3rd data frame. The pointer in the S&P field indicates the offset of the 3rd data frame from the S&P field. The 3rd data frame can again not completely be encoded in the 3rd symbol frame. The 4th data frame becomes completely encoded in the 4th symbol frame and is padded with fill bits, and so on. The pointer information in the S&P fields enables a receiver to recover from bit insertion and deletion errors.

To determine the overhead in framing bits per symbol, first let B_n be the number of bits that are encoded into the N_s symbols of the nth symbol frame. As mentioned above, the mean and standard deviation of the number of bits encoded per symbol dimension are R^h and σ_R^h, respectively, as given by (9) and (10). Then $B = N_s K R^h$ is the mean and $\sigma_B = \sqrt{N_s K}\, \sigma_R^h$ the standard deviation of B_n. For large N_s, the probability distribution of B_n will accurately be approximated by the Gaussian distribution

$$(13) \qquad \Pr(B_n = x) \cong \frac{1}{\sqrt{2\pi}\sigma_B} \exp\left(-\frac{(x-B)^2}{2\sigma_B^2} \right), \quad x = 0,1,2,3,\ldots$$

Next, let P_n be the pointer value in the S&P field of the nth symbol frame. The pointer values will remain bounded if $B > n_{sp} + N_b$. Equivalently, the average number of fill bits per frame, n_{fill}, is nonzero:

$$(14) \qquad n_{fill} = B - (n_{sp} + N_b) > 0.$$

Moreover, in a practical implementation the pointer values remain limited to the values that can be represented in the n_p-bit pointer subfield, i.e. $0 \leq P_n \leq 2^{n_p} - 1$. Parameters are chosen such that the probability of $P_n > 2^{n_p} - 1$ becomes negligible. From Fig. 7, one can verify the recursive relation

$$(15) \qquad P_n = \begin{cases} 0 & \text{if } n_{sp} + P_{n-1} + N_b \leq B_{n-1} \\ n_{sp} + P_{n-1} + N_b - B_{n-1} & \text{otherwise.} \end{cases}$$

The temporal evolution of the pointer probabilities then becomes

$$(16) \qquad \Pr(P_n = 0) = \sum_{x \geq 0} \Pr(P_{n-1} = x) \Pr(B_{n-1} \geq n_{sp} + N_b + x) \ ,$$

$$(17) \qquad \Pr(P_n = y) = \sum_{\substack{x \geq 0 \text{ and} \\ x \geq y - n_{sp} - N_b}} \Pr(P_{n-1} = x) \Pr(B_{n-1} = n_{sp} + N_b + x - y) ,$$

$$y = 1, 2, 3, \dots \ .$$

(equation (17) changed to fit within page margins)

The steady-state distribution $\Pr(P = x) = \Pr(P_{n \to \infty} = x)$ can be determined numerically (mathematically speaking, $\Pr(P = x)$ is the eigensolution of (16) and (17) associated with eigenvalue one). $\Pr(P = x)$ and $\Pr(P \geq x)$ are plotted in Fig. 8 for the following case.

- Lowest-energy 512-QAM, nominal rate R = 4 bit/dimension

- Huffman code design:
  $\Rightarrow R^h = 4.015$, $\sigma_R^h = 0.927$ bit/dimension; shaping gain $G_s^h = 1.412$ dB.

- Assume N_s = 512 QAM symbols/symbol, N_b = 4094 bit/data frame, n_{sp} = 12 (n_s=1, n_p=11)
  $\Rightarrow B = 4111.36$, $\sigma_B = 29.66$, $n_{fill} = 5.36$ bit/symbol frame.

The pointer field allows for a maximum pointer value of 2047. Fig. 6 shows that $Pr(P > 2047)$ is well below 10^{-10}. The pointer values exhibit a Paréto distribution, i.e., $\log(Pr(P \geq x))$ decreases linearly for large x.

A framing overhead of $(n_{sp} + n_{fill})/N_s = 0.034$ bit/QAM symbol is found, which is equivalent to an SNR penalty of 0.102 dB. The final net shaping gain becomes $1.412 - 0.102 = 1.310$ dB.

Based on the above mathematical foundation of Huffman shaping, in one embodiment of the invention, the method of the present invention may generally comprise two parts. The first is related to the design of the Huffman code to be employed on a given channel, and the second is related to the operation of the Huffman shaper in the transmitter. While the above mathematical foundation of Huffman shaping assumes a PAM implementation; extension to higher-dimensional modulation are also possible.

Fig. 9 illustrates one embodiment of the design of a Huffman code in accordance with the present invention. The modulation scheme is characterized by parameters M, α, and s (see (7) and accompanying text above) acquired in block 901, from which are derived the constellation levels $\{a_i; i = 0,1,...,M-1\}$ also in block 901. The probability p_i is then calculated for each a_i in step 903 for $i = 0,1,...,M-1$. Finally, a Huffman code for the symbols $\{a_i\}$ and their corresponding probabilities $\{p_i\}$ is constructed in block 905.

Fig. 10 is a block diagram of one embodiment of a communication system that operates in accordance with the method of present invention. Upon completion of the

construction of the Huffman code in Fig. 9, a Huffman shaper is employed. Referring to Fig. 10, Huffman shaper 1001 is loaded with information from a table similar to Table 1 above. The Huffman shaper information comprises one entry for each valid Huffman codeword and a corresponding entry for the channel symbol into which that Huffman codeword is mapped. The information is also sent to the receiver, using means available in the training procedure for the system. Then Huffman shaping proceeds during data transmission.

Specifically, referring again to Fig. 10, data source 1003 generates (typically binary, but this is not required) data symbols at an adjustable rate controlled by the Huffman shaper 1001. The data symbols are converted to pseudo-random form in a scrambler 1005. The Huffman shaper 1001 generally comprises two parts, namely, a Huffman parser 1007 and a mapper 1009. The Huffman parser 1007 accumulates outputs from the scrambler 1005, symbol by symbol (e.g., bit by bit), until it accumulates a valid Huffman codeword. This codeword forms the input to the mapper 1009. The mapper 1009 generates the channel symbol that corresponds to the Huffman codeword and passes the channel symbol to modulator 1011, under the control of the modulator clock 1013. The modulator clock 1013 defines the timing of the system. If required by the modulator clock 1013, the Huffman shaper 1001 controls the rate at which it accumulates output symbols from the scrambler 1005, in order to meet the demands of the modulator clock 1013.

Slicer/decision element 1015 maps the symbol received from the channel 1017 into its best estimate of the channel symbol transmitted by the remote transmitter. The Huffman encoder 1019 maps the estimated received channel symbol into a

Huffman codeword, which is passed to the descrambler 1021.
The descrambler 1021 inverts the operation of the scrambler
1005, and the resulting received sequence of data symbols is
passed to the user 1023.

The Huffman shaper 1001 is modeled as being able to
control the rate at which data are input to the shaper (see
reference numeral 1025 of Fig. 10). More colloquially, present-
day communication systems often operate in an environment
where a large buffer of data are available for transmission, and
data can be removed from that buffer at any rate appropriate for
the transmission medium. Therefore, ascribing an adjustable rate
capability to the Huffman shaper 1001 does not burden the
method of the present invention with functionality that is not
already present in practical situations.

As described above, a system that employs Huffman
shaping carries a variable number of bits per modulation symbol.
Therefore channel errors can introduce data in the receiver that is
incorrect bit-by-bit, and that actually may contain the wrong
number of bits as well. That is, referring to Fig. 10, if a channel
symbol different from the one introduced at the input to the
modulator 1011 is received at the output of the slicer/decision
element 1015, then both the bits and the number of bits passed to
the Huffman encoder 1019 may be incorrect. To compensate for
this potential effect, a framer/deframer may be introduced.

Fig. 11 is another embodiment of the design of a Huffman
code in accordance with the present invention, when a
framer/deframer is utilized. Again, a PAM implementation is
assumed, but extensions to higher-dimensional modulation are
also possible. Referring to Fig. 11, the modulation scheme is

characterized by parameters $M, \alpha, s, N_b, N_s, n_s$, and n_p acquired in

block 1001, from which are derived the constellation levels

$\{a_i ; i = 0,1,...,M - 1\}$ (block 1101). Parameters N_b, N_s, n_s, and n_p

define, respectively, the number of data bits, the number of

modulation symbols, the number of synch bits, and the number of

pointer bits in each symbol frame. The probability p_i then is

calculated for each a_i in block 1103 for $i = 0,1,...,M - 1$. Finally,

a Huffman code for the symbols $\{a_i\}$ and their corresponding

probabilities $\{p_i\}$ is constructed in block 1105.

Fig. 12 is a block diagram of another embodiment of a

communication system that operates in accordance with the

method of present invention, utilizing a framer/deframer. Upon

completion of the construction of the Huffman code in Fig. 11, a

Huffman shaper is employed. Referring to Fig. 12, Huffman

shaper 1201 is loaded with information from a table similar to

Table 1 above. The Huffman shaper information consists of one

entry for each valid Huffman codeword and a corresponding entry

for the channel symbol into which that Huffman codeword is

mapped. A framer 1203 is loaded with parameters N_b, N_s, n_s, and

n_p. The information is also sent to the receiver using means

available in the training procedure for the system. In the receiver

a deframer 1205 is loaded with the same parameters, N_b, N_s, n_s,

and n_p. Then Huffman shaping proceeds during data

transmission.

Specifically, referring to Fig. 12, data source 1207

generates data symbols at an adjustable rate controlled by the

Huffman shaper 1201. The data symbols are converted to

pseudo-random form in a scrambler 1209. The scrambler 1209

output is collected in the framer 1203, which arranges transmitted data in groups of N_b bits per symbol frame, N_s modulation symbols per symbol frame, n_s synch bits per frame and n_p pointer bits per frame as discussed above. The Huffman shaper 1201 generally comprises of two parts, a Huffman parser 1211 and the mapper 1213. The Huffman parser 1211 accumulates outputs from the framer 1203, symbol by symbol, until it accumulates a valid Huffman codeword. This codeword forms the input to the mapper 1213. The mapper 1213 generates the channel symbol that corresponds to the Huffman codeword and passes the channel symbol to the modulator 1215 under the control of the modulator clock 1217. The modulator clock 1217 defines the timing of the system. If required by the modulator clock 1217, the Huffman shaper 1201 controls the rate at which it accumulates output symbols from the scrambler 1209 in order to meet the demands of the modulator clock 1217 (see reference numeral 1218 in Fig.12).

The slicer/decision element 1219 maps the symbol received from the channel 1221 into its best estimate of the channel symbol transmitted by the remote transmitter. The Huffman encoder 1223 maps the estimated received channel symbol into a Huffman codeword. In this embodiment, switch 1225 is in position A. The deframer 1205 is able to distinguish individual received modulation symbols by means of the demodulator clock 1227 signal from the demodulator 1229. It uses the received symbol frame as well as the synch and pointer bits to construct a serial data stream corresponding to the output of the scrambler 1209. This output is passed to the descrambler 1231, which inverts the operation of the scrambler 1209, and the resulting received sequence of data symbols is passed to the user 1233.

In still another embodiment of the invention, the Huffman code constructed in a slightly modified fashion. This embodiment uses a one-dimensional form of the Huffman code described above. Specifically, a Huffman code is constructed for only the positive modulation symbols. After a Huffman code word has been collected in the transmitter by the Huffman decoder, the decoder uses its next input bit to define the sign of the modulation symbol corresponding to the collected Huffman code word. An inverse procedure is applied in the receiver. Again, a PAM implementation is assumed, but extension to higher-dimensional modulation is also possible.

Referring to Fig. 11, the modulation scheme is characterized by parameters $M, \alpha, s, N_b, N_s, n_s$, and n_p acquired in block 1101, from which are derived the constellation levels $\{a_i; i = 0,1,...,M-1\}$ (block 1101). Parameters N_b, N_s, n_s, and n_p define, respectively, the number of data bits, the number of modulation symbols, the number of synch bits, and the number of pointer bits in each symbol frame. The probability p_i is then calculated for each nonnegative a_i in block 1103 for $i = 0,1,...,M-1$. Finally, a Huffman code for the nonnegative symbols $\{a_i\}$ and their corresponding probabilities $\{p_i\}$ is constructed in block 1105.

Upon completion of the construction of the Huffman code in Fig. 11, a Huffman shaper is employed. Referring to Fig. 12, Huffman shaper 1201 is loaded with information from a table similar to Table 1 above. The Huffman shaper information consists of one entry for each valid Huffman codeword and a corresponding entry for the channel symbol into which that Huffman codeword is mapped. The framer 1203 is loaded with

parameters N_b, N_s, n_s, and n_p. The information is also sent to the
receiver using means available in the training procedure for the
system. In the receiver the deframer 1205 is loaded with the
same parameters, N_b, N_s, n_s, and n_p. Then Huffman shaping
proceeds during data transmission.

Specifically, data source 1207 generates data symbols at an
adjustable rate controlled by the Huffman shaper 1201. The data
symbols are converted to pseudo-random form in scrambler 1209.
The scrambler 1209 output is collected in the framer 1203, which
arranges transmitted data in groups of N_b bits per symbol frame,
N_s modulation symbols per symbol frame, n_s synch bits per
frame and n_p pointer bits per frame, as discussed above. The
Huffman shaper 1201 generally comprises two parts, the Huffman
parser 1211 and the mapper 1213. The Huffman parser 1211
accumulates outputs from the framer 1203, symbol by symbol,
until it accumulates a valid Huffman codeword. The Huffman
parser 1211 then accumulates one additional input bit and
appends it to the Huffman codeword. This Huffman codeword
with the appended bit forms the input to the mapper 1213. The
mapper 1213 generates the channel symbol that corresponds to
the Huffman codeword, and uses the appended bit to define the
sign of the channel symbol. It then passes the channel symbol to
the modulator 1215 under the control of the modulator clock
1217.

The slicer/decision element 1219 maps the magnitude of the
symbol received from the channel 1221 into its best estimate of the
magnitude of the channel symbol transmitted by the remote transmitter.
It also estimates the sign of the received symbol. The channel symbol
magnitude is passed to the Huffman encoder 1223, which maps the

estimated received channel symbol magnitude into a Huffman codeword
and presents the output at the A input of switch 1225. The sign of the
received symbol is presented at the B input of switch 1225 by means of
connection sign information 1235. Switch 1225, normally in the A
position; is switched to the B position after each received Huffman code
word, in order to accept the sign information 1235 from the
slicer/decision element 1219. The deframer 1205 is able to distinguish
individual received modulation symbols by means of the demodulator
clock 1227 signal from the demodulator 1229. It uses the received
symbol frame as well as the synch and pointer bits to construct a serial
data stream corresponding to the output of the scrambler 1209. This
output is passed to the descrambler 1231, which inverts the operation of
the scrambler 1209, and the resulting received sequence of data symbols
is passed to the user 1233.

Fig. 13 illustrates one operation of a system that employs
Huffman shaping in accordance with the present invention.
A transmitter 1301 accepts user data (block 1303). The
transmitter 1301 may also perform a framing operation (1307) to
provide a means to recover from possible errors that may be
introduced in the channel.

The transmitter 1301 then implements Huffman shaping.
Specifically, the transmitter 1301 accumulates source data until a
Huffman codeword is recognized (block 1309), and then maps
the resulting Huffman codeword into a channel symbol (block
1311). The transmitter then performs a modulation operation
(block 1313), which optionally includes sequence coding to
increase the signal distances between permitted symbol
sequences. Finally, the modulated signal is applied to the input
of the communications channel (block 1315).

The receiver 1317 accepts the received signal from the
channel output (block 1319), and demodulates it (block 1321).

Demodulation generally includes such operations as timing tracking and equalization. The received signal is then subjected to a decision operation, which may optionally include sequence decoding (block 1323). The Huffman shaping (blocks 1309 and 1311) is inverted by applying the received signal to the input of a Huffman encoder (block 1325). The receiver 1317 then performs a deframing operation (block 1327), and communicates the received data to the user (block 1331).

Based on the foregoing discussion, it should be apparent that in one embodiment of the invention, once data is received from a data source, the sequence of binary data bits is randomized by a scrambling operation and bits are mapped into channel symbols such that the channel symbols occur with a probability distribution suitable for achieving shaping gain. This is accomplished by accumulating scrambled data bits until a Huffman codeword is recognized, at which time the Huffman codeword is mapped into a channel symbol. Then the channel symbol is applied to the input of a communication channel. The probability of recognizing in the scrambled data sequence a particular Huffman codeword of length L bits is 2^{-L}. Hence, the channel symbol associated with that particular Huffman codeword will be transmitted with probability 2^{-L}. Note that this channel encoding operation via Huffman codes corresponds in the field of source coding to Huffman decoding.

In one embodiment of the invention, the channel encoding operation described above is performed in combination with a framing operation to achieve transmission of data at a constant rate. In addition, channel symbols can be modulated in various ways before they are applied to the input of the communication channel.

Next, on the receiver side of the communication channel a channel symbol is obtained at the demodulator output. The channel symbol is converted into the corresponding Huffman codeword. The sequence of bits represented by concatenated Huffman codewords is descrambled and delivered to the data sink. The described channel decoding operation corresponds in the field of source coding to Huffman encoding.

In one embodiment of the invention, a deframing operation is performed, which provides for data delivery to the data sink at constant rate. In addition, the deframing operation limits the effect of channel demodulation errors, which can cause a temporal shift of the received binary data sequence. This shift can occur when a channel symbol is erroneously decoded whose associated Huffman codeword differs in length from the Huffman codeword associated with the correct channel symbol.

The method of the present invention results in a symbol constellation and a probability distribution of symbols in this constellation that exhibits a shaping gain of greater than 1 dB. The shaping gain may be, for example, 1.35 dB or 1.5 dB, depending on the specific design. More specifically, for PAM constellations, shaping gains of up to ≈ 1.35 dB are achieved for some rates. For QAM constellations, shaping gains within 0.1 dB from the ultimate shaping gain are consistently obtained for rates of >3 bit per dimension.

In general, a communication system according to the present invention comprises a communication node that performs a Huffman decoding operation to generate channel symbols with a desired probability distribution.

Many modifications and variations of the present invention are possible in light of the above teachings. Thus, it is to be

understood that, within the scope of the appended claims, the invention may be practiced otherwise than as described hereinabove.

What is claimed and desired to be secured by Letters Patent is: